

| | |
|-----------|----------------|
| APPROVED | O.G. FIG. |
| BY | CLASS SUBCLASS |
| DRAFTSMAN | |

Atty: Dkt. No. 004-5723
1st Inventor: David L. Detlefs
Attorney: David W. O'Brien, (512) 347-9030

Sheet 1/4

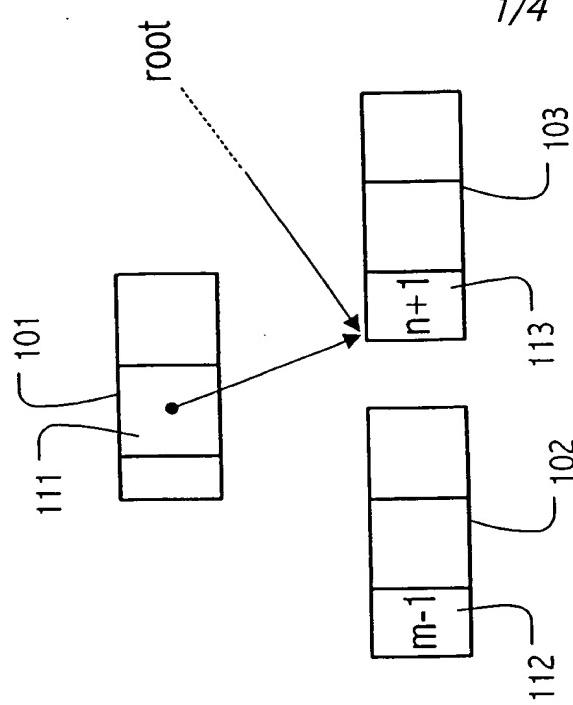


FIG 1B

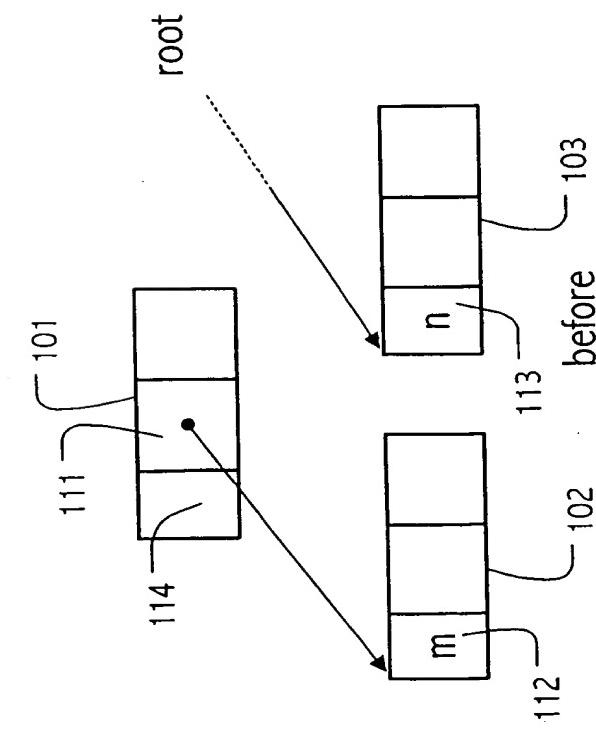
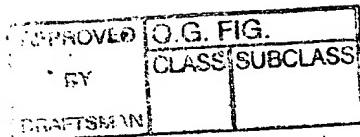


FIG. 1A



Atty. Dkt. No. 004-5723
1st Inventor: David L. Detlefs
Attorney: David W. O'Brien, (512) 347-9030

Sheet 2/4

2/4

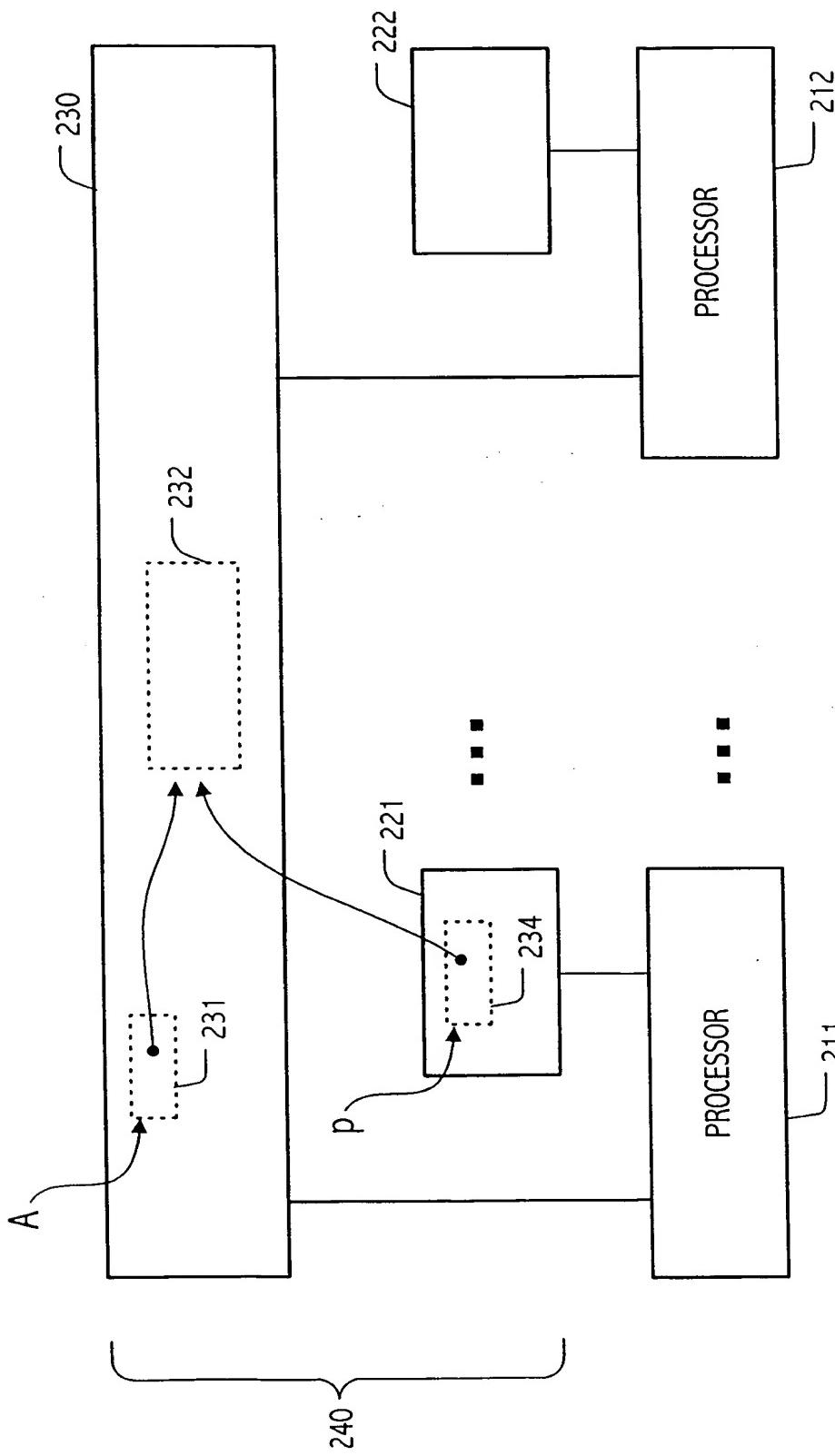


FIG. 2

O.G. FIG.
CLASS SUBCLASS

Atty. Dkt. No. 004-5723
1st Inventor: David L. Detlefs
Attorney: David W. O'Brien, (512) 347-9030

Sheet 3/4

3/4

```

class SNode {
1   class SNode *L, *R;  valtype V;
2   SNode() {};
};

class Snark {
3   SNode *Dummy, *LeftHat, *RightHat;
4   Snark() {
5     Dummy = new SNode;
6     Dummy->L = Dummy;
7     Dummy->R = Dummy;
8     LeftHat = Dummy;
9     RightHat = Dummy;
};

10  valtype pushRight(valtype v);
11  valtype pushLeft(valtype v);
12  valtype popRight();
13  valtype popLeft();
};

class SNode {
31  class SNode *L, *R;  valtype V;  long rc;
32  SNode() : L(Null), R(Null), rc(1) {};
};

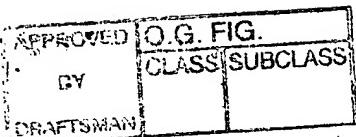
class Snark {
33  SNode *Dummy, *LeftHat, *RightHat;
34  Snark() : Dummy(Null),
35    LeftHat(Null), RightHat(Null) {
36    LFRCSAlloc(&Dummy, new SNode);
37    LFRCSStore(&Dummy->L, Null);
38    LFRCSStore(&Dummy->R, Null);
39    LFRCSStore(&LeftHat, Dummy);
40    ~Snark() {
41      while(popLeft()!=EMPTYval);
42      LFRCSStore(&Dummy, Null);
43      LFRCSStore(&LeftHat, Null);
44      LFRCSStore(&RightHat, Null);
45      valtype pushRight(valtype v);
46      valtype pushLeft(valtype v);
47      valtype popRight();
48      valtype popLeft();
};

10  valtype pushRight(valtype v);
11  valtype pushLeft(valtype v);
12  valtype popRight();
13  valtype popLeft();
};

```

FIG. 3A

FIG. 4A



Atty. Dkt. No. 004-5723
1st Inventor: David L. Detlefs
Attorney: David W. O'Brien, (512) 347-9030

Sheet 4/4

```

valtype Snark::pushRight(valtype v) {
    49 SNode *nd = new SNode;
    50 SNode *rh = Null, *rhR = Null, *lh = Null;
    51 if (nd == Null) {
        52 LFRCDestroy(rhR, nd, rh, lh);
        53 return FULLval;
    }
    54 LFRCStore(&nd→R, Dummy);
    55 nd→V = v;
    56 while (true) {
        57 LFRCLoad(&RightHat, &rh);
        58 LFRCLoad(&rh→R, &rhR);
        59 if (rhR == Null) {
            60 LFRCStore(&nd→L, Dummy);
            61 LFRCLoad(&LeftHat, &lh);
            62 if (DCAS(&RightHat, &LeftHat, rh, lh, nd, nd)) {
                63 LFRCDCAS(rhR, nd, rh, lh);
                64 return OKval;
            }
        }
    }
    65 LFRCStore(&nd→L, rh);
    66 if (DCAS(&RightHat, &rh→R, rh, rhR, nd, nd)) {
        67 LFRCDCAS(rhR, nd, rh, lh);
        68 return OKval;
    }
}

```

FIG 3B

FIG. 4B